Assignment 3

Sriyaank Vadlamani

2023-03-24

```
library(rpart)
library(geosphere)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(rpart.plot)
library(stringr)
library(moderndive)
```

```
set.seed(32523)
```

Datasets

```
train <- read.csv("train_data.csv")
test <- read.csv("test_data.csv")

train <- na.omit(train)
test <- na.omit(test)
```

Getting Variables Ready

Adding Distance from JDF and Distance from Broadway columns

```
jfk <- matrix(c( -73.7781, 40.6413), nrow=1) # uses latitude and longitude of JFK airport
broadway <- matrix(c(-73.9747, 40.7908), nrow=1) # uses latitude and longitude of broadway

# The distances are divided by 1000 to avoid scientific notation on decision tree
train$distance_jfk <- distGeo(jfk, matrix(c(train$longitude, train$latitude), ncol=2)) / 1000
test$distance_jfk <- distGeo(jfk, matrix(c(test$longitude, test$latitude), ncol=2)) / 1000

train$distance_broadway <- distGeo(broadway, matrix(c(train$longitude, train$latitude), ncol=2)) / 1000
test$distance_broadway <- distGeo(broadway, matrix(c(test$longitude, test$latitude), ncol=2))/ 1000
```

Adding has_crime column

```
crime_data <- read.csv("NYC_Crime_Statistics.csv")
crime_dict <- with(crime_data, setNames(Zip.Codes, TOTAL.SEVEN.MAJOR.FELONY.OFFENSES))

train$zipcode <- str_sub(train$Location, -20, -16)
train$zipcode <- as.integer(train$zipcode)
```

```
## Warning: NAs introduced by coercion
```

```
train$has_crime <- ifelse(any(crime_data == train$zipcode), TRUE, FALSE)

train$has_crime <- train$zipcode %in% crime_data$Zip.Codes
```

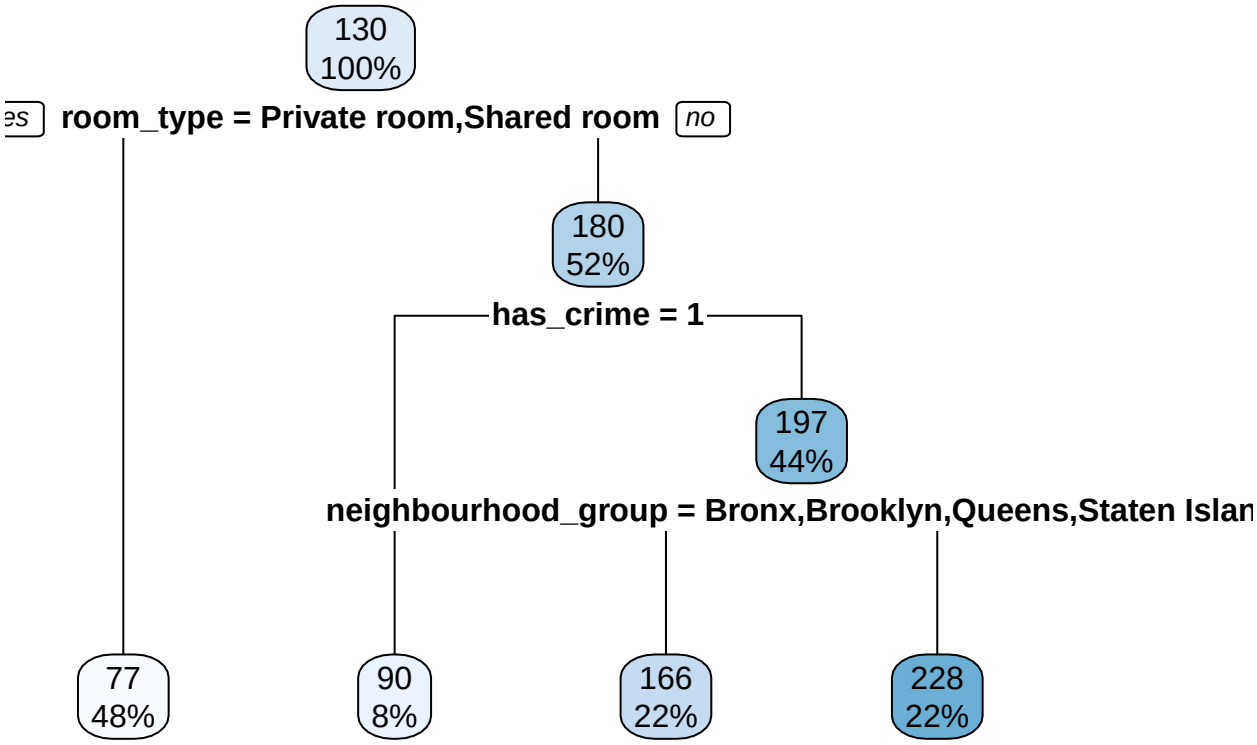
Making Decision Trees

Test #1: Initial Test

```
train1 <- train %>% select(neighbourhood_group, room_type, distance_jfk, distance_broadway, has_crime, p
fit1 <- rpart(price ~ ., data = train1, method = "anova")
fit1
```

```
## n= 31140
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##  1) root 31140 1158834000 130.49360
##    2) room_type=Private room,Shared room 14871  322201700  76.55289 *
##    3) room_type=Entire home/apt 16269  753812600 179.79930
##      6) has_crime>=0.5 2590   15797720  90.35869 *
##      7) has_crime< 0.5 13679  713372900 196.73400
##       14) neighbourhood_group=Bronx,Brooklyn,Queens,Staten Island 6896  225221800 165.91110 *
##       15) neighbourhood_group=Manhattan 6783  474938800 228.07050 *
```

```
rpart.plot(fit1)
```



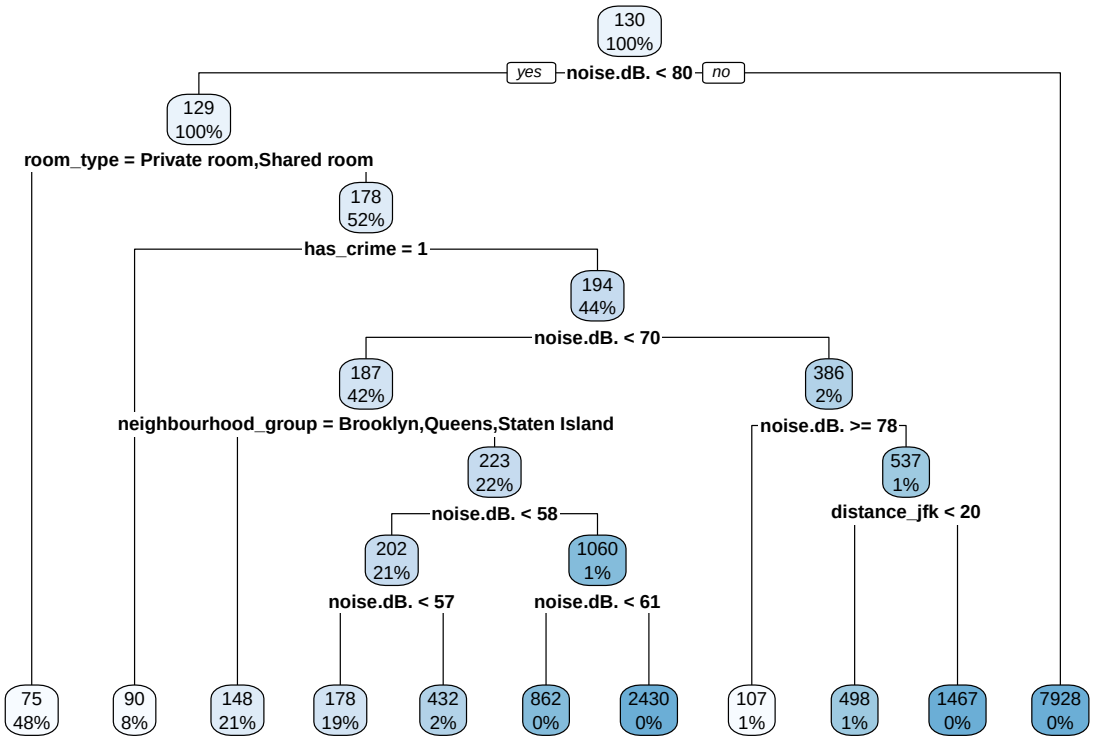
Test #2: More variables

```
train2 <- train %>% select(neighbourhood_group, room_type, distance_jfk, minimum_nights, number_of_revi
fit2 <- rpart(price ~ ., data = train2, method = "anova")
fit2
```

```
## n= 31140
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##   1) root 31140 1158834000.0  130.49360
##     2) noise.dB.< 80.26418 31133  699393000.0  128.74040
##       4) room_type=Private room,Shared room 14869  168618900.0   75.38631 *
##       5) room_type=Entire home/apt 16264  449750700.0  177.51810
##        10) has_crime>=0.5 2590   15797720.0   90.35869 *
##        11) has_crime< 0.5 13674  410550600.0  194.02710
##          22) noise.dB.< 69.55537 13174  291652100.0  186.72290
##            44) neighbourhood_group=Brooklyn,Queens,Staten Island 6396   31383650.0  147.98060 *
##            45) neighbourhood_group=Manhattan 6778  241609200.0  223.28160
##              90) noise.dB.< 57.55428 6611   63127710.0  202.15530
##               180) noise.dB.< 56.55428 5971   21782430.0  177.53420 *
##               181) noise.dB.>=56.55428 640    3955824.0  431.86250 *
##              91) noise.dB.>=57.55428 167   58724960.0 1059.60500
##               182) noise.dB.< 60.55428 146    5106622.0  862.47260 *
##               183) noise.dB.>=60.55428 21    8498751.0 2430.14300 *
##          23) noise.dB.>=69.55537 500   99676980.0  386.47800
##            46) noise.dB.>=77.97407 175     270329.7  107.11430 *
##            47) noise.dB.< 77.97407 325   78394780.0  536.90460
##              94) distance_jfk< 20.20958 312   16903920.0  498.15710 *
```

```
##              95) distance_jfk>=20.20958 13   49780140.0 1466.84600 *
##     3) noise.dB.>=80.26418 7   33706000.0 7928.28600 *
```

```
rpart.plot(fit2)
```



Test #3:

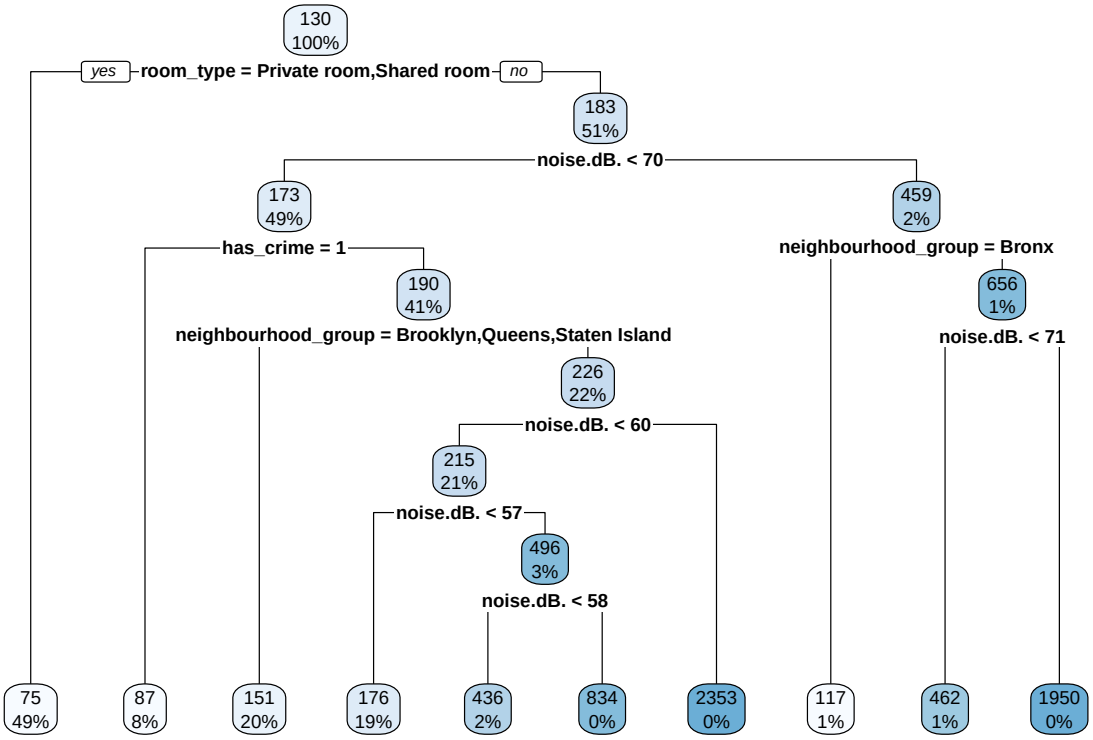
Final test, more cleaned up

```
train3 <- rep_sample_n(train, size=7703, replace = FALSE) # Using sample of same size as test dataset t
fit3 <- rpart(price ~ room_type + noise.dB. + neighbourhood_group + has_crime,data = train3, method = "a
fit3
```

```
## n= 7703
##
## node), split, n, deviance, yval
##       * denotes terminal node
##
##   1) root 7703 288133400.0  130.46380
##     2) room_type=Private room,Shared room 3748  19456640.0   74.63901 *
##     3) room_type=Entire home/apt 3955 245927500.0  183.36690
##       6) noise.dB.< 69.55537 3810  95155780.0  172.86980
##        12) has_crime>=0.5 620   2611583.0   87.03871 *
##        13) has_crime< 0.5 3190  87088930.0  189.55170
##          26) neighbourhood_group=Brooklyn,Queens,Staten Island 1533  12668050.0  150.53160 *
##          27) neighbourhood_group=Manhattan 1657  69927360.0  225.65180
##            54) noise.dB.< 59.55428 1649  29351950.0  215.33290
##             108) noise.dB.< 56.55428 1448   5328907.0  176.43230 *
##             109) noise.dB.>=56.55428 201   6046513.0  495.57210
##               218) noise.dB.< 57.55428 171   1178622.0  436.12280 *
##               219) noise.dB.>=57.55428 30    818731.4  834.43330 *
##            55) noise.dB.>=59.55428 8   4207706.0 2352.62500 *
##       7) noise.dB.>=69.55537 145 139320800.0  459.18620
##        14) neighbourhood_group=Bronx 53    226950.8  116.71700 *
```

```
##         15) neighbourhood_group=Brooklyn,Manhattan 92 129296700.0  656.47830
##           30) noise.dB.< 70.55646 80  21791350.0  462.45000 *
##           31) noise.dB.>=70.55646 12  84415220.0 1950.00000 *
```

```
rpart.plot(fit3)
```



Predict the test data

Add has_crime to test

```
crime_data <- read.csv("NYC_Crime_Statistics.csv")
crime_dict <- with(crime_data, setNames(Zip.Codes, TOTAL.SEVEN.MAJOR.FELONY.OFFENSES))

test$zipcode <- str_sub(test$Location, -20, -16)
test$zipcode <- as.integer(test$zipcode)
```

```
## Warning: NAs introduced by coercion
```

```
test$has_crime <- ifelse(any(crime_data == test$zipcode), TRUE, FALSE)

test$has_crime <- test$zipcode %in% crime_data$Zip.Codes
```

Prediction time!

```
test$price <- predict(fit3)
```